

Current Status: I'm available for any 6 to 12 month remote contracts from 24 to 40 hours per week. Check out my website www.emicros.com as well as search [udemy.com](https://www.udemy.com) for "Ron Russ" to see my CAN and LIN courses.

As a highly trained Embedded Software Engineer with extensive experience and skills in all aspects of embedded software engineering in the Automotive segment, I can produce high quality and robust embedded software. With 43 total years of technical experience in Embedded Software Development, 34 years devoted to the Automotive market, will allow me to positively influence your software organization immediately. The automotive embedded software engineering experience started at Ford in 1990 writing assembly software for a high volume Instrument Cluster for a Ford/VW European mini-van as well as a Mazda pickup truck. Next was a C based Jaguar Instrument Cluster featuring a Controller Area Network and SCP communication network. When Ford spun our division off as Visteon I worked on algorithms and software for our Stepper Motor Manager, Spatial Infrared Gesture detection, capacitive touchscreen software, and Push-With-Intent processing. Other experience include 8 months at IYU a Chinese Company, C/C++ at Harman working on name brand audio amps, and then LIN based Overhead Consoles at Flex (formerly Flextronics). I've worked with multi-national OEM customers in all phases of the software product development process from capturing customer requirements to launching automotive products with embedded software. Interfaced with customers in the US (Ford and Honda), Japan (Honda and Mazda), Germany (VW), and England (Jaguar), on-site technical reviews in Brazil, and supported manufacturing sites overseas.

EXPERIENCE

12/2022 – 8/2023

PRINCIPAL SOFTWARE ENGINEER, FLEX

Contract assignment with Flex supporting current model Over Head Console for Stellantis and GM. The Stellantis OHC embedded software maintenance consists of adding new variants from the current ones while developing one basic version with end-of-line configurability. While working on the LIN (Local Interconnect Network) based OHC's developed a low-cost LIN interface tool with both USB and Bluetooth interfaces as well as a UDEMY LIN course. Provided GM generation 1.5, 2, and 3 OHC support. Worked with Analyzer for LIN communications.

4/2020 – 12/2022

AVAILABLE FOR NEW ASSIGNMENTS

After being laid off from Harman due to the pandemic I decided to take the time off but kept my software skills sharp through software and hardware design via my www.emicros.com website.

10/2017 – 4/2020

PRINCIPAL SOFTWARE ENGINEER, HARMAN

Responsible for design and implementation of the Critical Chime Producer feature for model year 2022 premium branded AKG amplifier based on TI Jacinto multi-core processor. I wrote a Visual Studio C# program to convert critical chimes files, IDs, and playback configurations into a format stored in the flash file system. Playback configuration consisted of single or multiple times as well as linkage to another chime. Software written in C++. Used Vector CANalyzer for triggering chimes for initial development and later designed a Teensy 3.5 based CAN board for exercising all chimes and collecting results via both the onboard USB and SSD card. Experience with the A2B

was also gained on this amplifier. On the premium Mark Levinson branded amplifier using an NXP S32 micro running the MicroC/OS kernel (μ C/OS-II) worked on low level SPI and UART drivers as well as power up/down feature. Used Atlassian Jira/Confluence/SourceTree/Bitbucket for Agile software process.

1/2017 – 8/2017

EMBEDDED SOFTWARE ENGINEER, IYU AUTOMOTIVE

Developed embedded software for DC to AC Inverter and Tire Pressure Monitor System. Benchmarked various Overhead Consoles from competitors. IYU Automotive is a Chinese injection molding and stamping company and had plans to expand in the North American automotive market but the anticipated growth did not occur to support a dedicated embedded software engineer.

11/1990 – 12/2016

EMBEDDED SOFTWARE ENGINEER, FORD/VISTEON

FORD – High Volume Instrument Clusters

- Hired by Ford in 11/1990 as an Embedded Software Engineer specifically to work on the upcoming hybrid instrument cluster the 1995.5 VW/Ford European Instrument Cluster. This ECU consisted of 3 versions from a basic 4 air-core gauge cluster to an added warnings module to one with a trip computer. This product used a Motorola 68HC11E20 micro programmed in assembler. The minivan for Ford was the Galaxy and the VW version was the Sharron. This program required numerous trips to Germany to work with the VW engineers to review requirements, perform unit and vehicle testing, and final signoff.
- The next program was the 1997 Jaguar High-End Instrument Cluster that contained 6 Switec stepper motors, an LCD message center, and two in-vehicle networks; CAN (Controller Area Network) and Ford's SCP (Standard Corporate Protocol). The product used a Motorola 68HC11KA2 microcontroller and the software was written in a mixture of C and assembler. Numerous on-site meetings in England with the Jaguar engineers occurred also.
- Designed and developed the Motorola 68HC05 software in assembly for both the 1998 Honda Instrument Cluster and 1998.5 Mazda J97 Cluster. Both of these clusters were air-core based having 3 or 4 gauges with an LCD.
- The above instrument clusters were high-volume and were "bare metal" operating systems.

VISTEON – Stepper Motor Subject Matter Expert

- In 2000 Ford spun off the Automotive Components Division as Visteon.
- In the late 1990's and early 2000's the emphasis was on expanding the software development process to have local expert responsible for specific bookshelf software packages.
- The Stepper Motor Core Software driver package for interfacing either Switec or MMT stepper motors to Driver Information products was assigned to me. Tasks included generating software requirements, design specification, implementation, unit test, and user documentation as well as holding reviews and following the Visteon Software Process. The packages were platform based (i.e. different micros) and eventually as our off-shore software development presence expanded I mentored new software engineers and eventually transferred the responsibility to them.

VISTEON – Advanced Technologies

- After doing high volume production software I was drafted to design and implement software for some cutting-edge technologies. The first was the Hidden Touch & Dual

OLED showcase properties first shown at the Consumer Electronics Show. Capacitive touch and capacitive push with intent used on these properties. The push with intent allows drivers to touch the screen and move around then with a slight push select an item.

- Then we added what I coined VIRTU which stood for Visteon InfraRed Technology Unit. This is an infrared detection system for Spatial Gesture Recognition providing both driver and passenger gesture recognition featuring left and right swipes, hand approach, hand tracking, rotation, and selection. This was the last program I worked on at Visteon before accepting an early buyout at the end of 2016 so the status of these technologies are unknown to me.

1/1986 – 11/1990

ELECTRICAL & SOFTWARE ENGINEER, RAYCON

Designed hardware and software for Electrical Discharge Machines. Designed Pulse Generator, Gap Sense Protection, and 8051 MicroModule/Network boards. Designed system software for RE35 EDM and IBM based operating system for RP200. Tango software was used for both schematics capture as well as printed circuit board layout.

EDUCATION

12/1994

MASTERS OF SCIENCE IN ELECTRONICS & COMPUTER CONTROL, WAYNE STATE UNIVERSITY

GPA 4.0/4.0

6/1981

BACHELOR OF SCIENCE IN ELECTRICAL ENGINEERING, LAWRENCE INSTITUTE OF TECHNOLOGY

GPA 3.54/4.0

SKILLS

LANGUAGES	Proficient in Assembler, Embedded C, C#, Basic, and C++
MICROCONTROLLERS	Renesas V850, MicroChip PIC 16F876, TI Jacinto 6, Arm, NXP(Freescale) from 68HC05 to 68HC11/12 to Rainbow/Spectrum (Power PC based) to IMX6 to S32K148
PLATFORMS	Provided software subsystems for Kepler, Newton, and other Platforms based on the Driver Information Kernel (in house operating system)
COMMUNICATION	Worked with and designed software for CAN, I2C, SPI, LIN, and UART based interfaces. The 1996 Jaguar X100 cluster was the first CAN based instrument cluster that also contained an SCP vehicle network and was the first cluster written in C. Worked with LIN based B515 Center Stack prototype.

**INTEGRATED
DEVELOPMENT
ENVIRONMENTS,
TOOLS, & TRAINING**

Software development using IDE's such as CodeWarrior IDE, GreenHills Multi, Atmel Touch Studio, and Silicon Labs Simplicity.
Use software quality tools QAC for static analysis, trained in VectorCast for C testing on a PC, and used ClearQuest for defect tracking.
Used PVCS, ClearCase, and RTC for archiving software
Uses C# and a serial UART link to the embedded system for development, testing, calibration, and prove-out.

OPERATING SYSTEMS

Experience in MicroC/OS-II. Exposure to Linux and AutoSAR.

PATENTS

[9,673,740 Determining a rotor offset](#)

[9,013,133 Method and apparatus for stepper motor stall detection \(No Twitch Method\)](#)

[7,880,423 Method and apparatus for stepper motor stall detection](#)

[7,812,562 Method and apparatus for high speed stepper motor stall detection](#)

[6,853,162 Re-zeroing of a stepper motor without noise or movement](#)

[6,219,624 Synchronous timer vehicle speed measurement](#)

[5,561,374 Method for displaying a vehicle speed measurement with improved response](#)
